UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/601,521	06/24/2003	Yue-Sun Kuo	KUOY3001/EM	7539

23364 7590 12/01/2006

BACON & THOMAS, PLLC
625 SLATERS LANE
FOURTH FLOOR
ALEXANDRIA, VA 22314

EXAMINER	
STEELMAN, MARY J	

ART UNIT	PAPER NUMBER
2191	

DATE MAILED: 12/01/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/601,521	KUO ET AL.	
	Examiner	Art Unit	
	Mary J. Steelman	2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *24 June 2003*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-18* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-18* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This Office Action is in response to application received 06/24/2003. Claims 1-18 are pending.

Claim Objections

2. Claim 1 is objected to because Examiner fails to understand line 7, "a list of candidate third element to be alerted to user..."

3. Claim 7 is objected to because Examiner fails to understand line 19, "a list of candidate third element to be alerted to user..."

4. Claim 13, line 2 recites "usef", should be –user-- Replace 'f' with 'r'.

5. Examiner recognizes that terminology, converted in scanning for publication, in for claims 1-18, such as ver-tline, vertline, epsilon, etc, is confusing. For purposes of presenting claim limitations in this office action, the scanned version is used. Claims limitations remain as originally submitted.

Claim Rejections - 35 USC § 101

6. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

7. Claims 1-6 and 13-18 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter, software per se. Such software, lacking storage on a

suitable computer-readable medium are not able to realize any functionality and are thus not statutory.

8. Claims 7-12 are rejected under 35 U.S.C. 101. Claims 7-12 fail to provide a practical application, as there is no physical transformation, useful, concrete or tangible result.

See MPEP 2106.02 [R-5] Mathematical Algorithms

Claims to processes that do nothing more than solve mathematical problems or manipulate abstract ideas or concepts are complex to analyze and are addressed herein. If the "acts" of a claimed process manipulate only numbers, abstract concepts or ideas, or signals representing any of the foregoing, the acts are not being applied to appropriate subject matter. Gottschalk v. Benson, 409 U.S. 63, 71 - 72, 175 USPQ 673, 676 (1972). Thus, a process consisting solely of mathematical operations, i.e., converting one set of numbers into another set of numbers, does not manipulate appropriate subject matter and thus cannot constitute a statutory process.

In practical terms, claims define nonstatutory processes if they:

– consist solely of mathematical operations without some claimed practical application (i.e., executing a "mathematical algorithm"); or

– simply manipulate abstract ideas, e.g., a bid (Schrader, 22 F.3d at 293-94, 30 USPQ2d at 1458-59) or a bubble hierarchy (Warmerdam, 33 F.3d at 1360, 31 USPQ2d at 1759), without some claimed practical application.

Claim Rejections - 35 USC § 103

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

10. Claims 1-18 are rejected under 35 U.S.C. 103(a) as being unpatentable over "Rita-an editor and user interface for manipulating structured documents" by D. D. Cowan, E. W. Mackie, and G. M. Pianosi and G. de V. Smit (hereinafter 'Rita') (1991), in view of "Regular Expressions into Finite Automata" by A. Bruggermann-Klein (1993) (hereinafter "Bruggermann-Klein").

An XML document editor to enable user to add or delete an element into a working document and to convert said working document into an XML document file;

See Rita, page 125, last paragraph, "Rita, an editor and user interface for structured documents"

See Rita, page 129, middle paragraph, "ability to edit both text and structure (add, delete, convert) during document creation or revision"

-characterized in that said XML document editor automatically generates in relating to two consecutive elements z.sub.i and z.sub.i+1 of said working document, wherein relation between said elements z.sub.i and z.sub.i+1 complies with document type definition (DTD) of said document, a list of candidate third element to be alerted to user;

See Rita, page 131, 3rd paragraph, "Only elements which can be legally inserted before the cursor are shown in the structure menu."

-wherein said third element z in said list makes relations between elements z.sub.i and z and between elements z and z.sub.i+1 complying with said DTD, after said element z is inserted between elements z.sub.i and z.sub.i+1.

Rita, page 135, 3rd paragraph, "First, a list of all of the potential document elements is created from the transitions of the FSA (finite state automata). Second, each of them is then tested to see if its inclusion causes the automaton to achieve a defined state and thus represents a valid insertion at this point in the document." Complying elements may be inserted between two existing elements.

Bruggermann-Klein disclosed formulas related to the above limitations for inserting elements into a document.

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to modify Rita, to include the formulas as provided by Bruggermann-Klein, because Rita disclosed (page 134, last paragraph) a FSA provides an equivalent definition of the document class description. Bruggermann-Klein provided support with an FSA of the type Glsuhkov automaton, used because (page 2, 4th paragraph) "states correspond to the occurrences of symbols in E and whose transitions connect positions that can be consecutive on a path through E."

Per claim 2:

-XML document editor determines whether relation between two consecutive elements comply with said DTD according to the following rule:

-suppose G is Glushkov Automaton of said document, z.sub.i is a state in G, 1.ltoreq.i.ltoreq.p-1, p.epsilon.N, .SIGMA.={z.sub.1, z.sub.2, . . . z.sub.p} is a sequence of states in G where z.sub.1=s, s is start state of G, z.sub.p=f, f is final state of G; if z.sub.i+1.epsilon.reachable(z.-sub.i), wherein reachable(z.sub.i) denote the set of states in G reachable from state z.sub.i, then the relation between z.sub.i and z.sub.i+1 is determined compliant with DTD of said document.

Rita, page 135, 5th paragraph, "Ths FSAs are used not only to determine which objects can be added at a given location in the document, but also to check the correctness of composite objects and the legality of all structural editing operations."

Rita, page 135, 7th paragraph, "The philosophy used in Rita is to provide the user with those options which are consistent with the document grammar so that mistakes are prevented. Thus the creation menu contains only legal document elements and is calculated dynamically from the finite automaton constructed for the object class of the parent of the current object."

Rita failed to explicitly disclose a Glushkov Automaton (a finite state automaton), however, Buggemann-Klein disclosed using the Glushkov automaton for transforming regular expressions (page 1, 1st paragraph) into a nondeterministic finite automaton (NFA) (page 2, 2nd paragraph).

Page 3, 1st paragraph, « In the SGML context, the only valid regular expressions are those for which the Glushkov automaton is deterministic.

Buggemann-Klein disclosed (page 4, last paragraph) a first set of positions (start), a last set of positions (final) and follow set of positions (reachable) that can follow position x in a path through E.

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to modify Rita, to include the formulas as provided by Bruggermann-Klein, because Rita disclosed (page 134, last paragraph) a FSA provides an equivalent definition of the document class description. Bruggermann-Klein provided support with an FSA of the type Glsuhkov automaton, used because (page 2, 4th paragraph) "states correspond to the occurrences of symbols in E and whose transitions connect positions that can be consecutive on a path through E."

Per claim 3:

-XML generates a cell C to include said candidate third element z according to the following rule and displays said candidates in a list:

-suppose (z.sub.i, z.sub.i+1)H, H denotes the set of edges in G, G is Glushkov Automaton of regular expression .SIGMA. corresponding to an element of said working document; further suppose .SIGMA. is a set to include states corresponding to all elements of G, A(E1) is the set of states in subexpression E1 to E, f-reachable(z.sub.i) denotes the set of states in G reachable from

z.sub.i through forward edges; if z.sub.i+1.epsilon.f-reachable(z.sub.i), then let C={z.epsilon.Z.vertline.- z.epsilon.f-reachable(z.sub.i) and z.sub.i+1.epsilon.f-reachable(z)}; if z.sub.i+1f-reachable(z.sub.i), then let E1* be the smallest iteration subexpression of E that covers both z.sub.i and z.sub.i+1, C={z.epsilon.A(E1).vertline.z.epsilon.f-reachable(z.sub.i) or z.sub.i+1.epsilon.f-reachable(z)}.

Buggemann-Klein disclosed (page 4, last paragraph) a follow set of positions (reachable) that can follow position x in a path through E. Buggemann-Klein disclosed, page 15, 4th paragraph, "Any non-empty path through ME is determined by a sequence of positions x1, ..., xn, n>= 1, which consists of a sequence of paths through MF. Because follow (F, last...the starting positions of those paths are uniquely determined." Denote the set of states reachable.

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to modify Rita, to include the formulas as provided by Bruggermann-Klein, because Rita disclosed (page 134, last paragraph) a FSA provides an equivalent definition of the document class description. Bruggermann-Klein provided support with an FSA of the type Glsuhkov automaton, used because (page 2, 4th paragraph) "states correspond to the occurrences of symbols in E and whose transitions connect positions that can be consecutive on a path through E."

Per claim 4:

-XML generates a cell C to include said candidate third element z according to the following rule and displays said candidates in a list: suppose (z.sub.i, z.sub.i+1).epsilon.H, H denotes the set of

edges in G, G is Glushkov Automaton of regular expression E corresponding to an element of said working document; further suppose L is a set to include states corresponding to all elements of G, A(E1) is the set of states in subexpression E1 to E, f-reachable(z.sub.i) denotes the set of states in G reachable from z.sub.i through forward edges; if (z.sub.i, z.sub.i+1) is a forward edge, let C={z.epsilon..SIGMA..vertline.z.epsilon.f-reachabl- e(z.sub.i) and z.sub.i+1.epsilon.f-reachable(z)} and: i) if z.sub.i.epsilon.last(E1*) for some iteration subexpression E1* of E and E1 is the largest one, then let C1={z.epsilon.A(E1).vertline.z.sub.i+1.ep- silon.f-reachable(z.sub.i)}, C=C.Arrow-up bold.C1; ii) If z.sub.i+1.epsilon.first(E2*) for some iteration subexpression E2* of E and E2 is the largest one, then let C2={z.epsilon.A(E2).vertline.z.epsilo-n.f-reachable(z.sub.i)} and C=CC2; if (z.sub.i, z.sub.i+1) is a backward edge, then let C=A(E3), wherein E3* is the largest iteration subexpression of E satisfying z.sub.i.epsilon.last(E3) and z.sub.i+1.epsilon.first(E3).

Buggemann-Klein disclosed (page 4, last paragraph) a follow set of positions (reachable) that can follow position x in a path through E. Buggemann-Klein disclosed, page 15, 4th paragraph, "Any non-empty path through ME is determined by a sequence of positions x1, ..., xn, n>= 1, which consists of a sequence of paths through MF. Because follow (F, last...the starting positions of those paths are uniquely determined." Denote the entire set of states reachable.

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to modify Rita, to include the formulas as provided by Bruggermann-Klein, because Rita disclosed (page 134, last paragraph) a FSA provides an equivalent definition of the document class description. Bruggermann-Klein provided support with an FSA of the type

Glsuhkov automaton, used because (page 2, 4th paragraph) "states correspond to the occurrences of symbols in E and whose transitions connect positions that can be consecutive on a path through E."

Per claim 5:

-XML document editor automatically generates a required element between element pair z.sub.i and z and element pair z and z.sub.i+1 after said third element z is inserted between element pair z.sub.i and z.sub.i+1 such that said working document is effective; wherein said requirement comprises articulation points between elements z.sub.i and z (and z and z.sub.i+1) in Glushkov Automaton G; i.e., states through which all paths between z.sub.i and z (and z and z.sub.i+1) shall pass.

Buggemann-Klein disclosed (page 4, last paragraph) a follow set of positions (reachable) that can follow position x in a path through E. Buggemann-Klein disclosed, page 15, 4th paragraph, "Any non-empty path through ME is determined by a sequence of positions x1, ..., xn, n>= 1, which consists of a sequence of paths through MF. Because follow (F, last...the starting positions of those paths are uniquely determined." Paths are routed through nodes in positions as required by the grammar rules.

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to modify Rita, to include the formulas as provided by Bruggermann-Klein, because Rita disclosed (page 134, last paragraph) a FSA provides an equivalent definition of the document class description. Bruggermann-Klein provided support with an FSA of the type

Glsuhkov automaton, used because (page 2, 4th paragraph) "states correspond to the occurrences of symbols in E and whose transitions connect positions that can be consecutive on a path through E."

Per claim 6:

-XML document editor automatically generates an element slot allowing user to add elements into said document, if no required element between element pair z.sub.i and z and element pair z and z.sub.i+1 is found after said third element z is inserted between element pair z.sub.i and z.sub.i+1 and if (z.sub.i, z)H((z, z.sub.i+1)H), wherein H denotes set of edges in G; and wherein said requirement comprises articulation points between elements z.sub.i and z (and z and z.sub.i+1) in Glushkov Automaton G; i.e., states through which all paths between z.sub.i and z (and z and z.sub.i+1) shall pass.

Rita disclosed adding elements into the document. See rejections addressed in claim 1 above.

Rita failed to disclose the Glushkov Automaton.

However, Buggemann-Klein disclosed, page 15, 4th paragraph, "Any non-empty path through ME is determined by a sequence of positions x1, …, xn, n>= 1, which consists of a sequence of paths through MF. Because follow (F, last…the starting positions of those paths are uniquely determined." Buggemann-Klein disclosed 'articulation points' through which a non-empty path will pass.

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to modify Rita, to include the formulas as provided by Bruggermann-Klein, because

Rita disclosed (page 134, last paragraph) a FSA provides an equivalent definition of the document class description. Bruggermann-Klein provided support with an FSA of the type Glsuhkov automaton, used because (page 2, 4th paragraph) "states correspond to the occurrences of symbols in E and whose transitions connect positions that can be consecutive on a path through E."

Per claim 7:

Method for editing an XML document using an XML document editor to enable user to add or delete an element into a working document and to convert said working document into an XML document file;

-characterized in that said method comprising enabling said XML document editor to automatically generate in relating to two consecutive elements z.sub.i and z.sub.i+1 of said working document, wherein relation between said elements z.sub.i and z.sub.i+1 complies with document type definition (DTD) of said document, a list of candidate third element to be alerted to user; wherein said third element z in said list makes relations between elements z.sub.i and z and between elements z and z.sub.i+1 complying with said DTD, after said element z is inserted between elements z.sub.i and z.sub.i+1.

See rejection of limitations addressed in claim 1 above.

Per claim 8:

-whether relation between two consecutive elements complies with said DTD is determined according to the following rule:

-suppose G is Glushkov Automaton of said document, z.sub.i is a state in G, 1.ltoreq.i.ltoreq.p-1, p.epsilon.N, .SIGMA.={z.sub.1, z.sub.2, . . . z.sub.p} is a sequence of states in G where z.sub.1=s, s is start state of G, z.sub.p=f, f is final state of G; if z.sub.i+1.epsilon.reachable(z.sub.i), wherein reachable(z.sub.i) denote the set of states in G reachable from state z.sub.i, then the relation between z.sub.i and z.sub.i+1 is determined compliant with DTD of said document.

See rejection of limitations addressed in claim 2 above.

Per claim 9:

-a cell C to include said candidate third element z is generated according to the following rule and displayed as a list:

-suppose (z.sub.i, z.sub.i+1)H, H denotes the set of edges in G, G is Glushkov Automaton of regular expression E corresponding to an element of said working document; further suppose .SIGMA. is a set to include states corresponding to all elements of G, A(E1) is the set of states in subexpression E1 to E, f-reachable(z.sub.i) denotes the set of states in G reachable from z.sub.i through forward edges; if z.sub.i+1.epsilon.f-reachable(z.sub.i), then let C={z.epsilon..SIGMA..vertline.z.epsilon.f-reachable(z.sub.i) and z.sub.i+1.epsilon.f-

reachable(z)}; if z.sub.i+1f-reachable(z.sub.i), then let E1* be the smallest iteration subexpression of E that covers both z.sub.i and z.sub.i+1, C={z.epsilon.A(E1).vertline.z.epsilon.f-reachable(- z.sub.i) or z.sub.i+1.epsilon.f-reachable(z)}. See rejection of limitations addressed in claim 3 above.

Per claim 10:

-wherein a cell C to include said candidate third element z is generated according to the following rule and displayed as a list: suppose (z.sub.i, z.sub.i+1).epsilon.H, H denotes the set of edges in G, G is Glushkov Automaton of regular expression E corresponding to an element of said working document; further suppose .SIGMA. is a set to include states corresponding to all elements of G, A(E1) is the set of states in subexpression E1 to E, f-reachable(z.sub.i) denotes the set of states in G reachable from z.sub.i through forward edges;

if (z.sub.i, z.sub.i+1) is a forward edge, let C={z.epsilon..SIGMA..vertline.z.epsilon.f-reachable(z.sub.i) and z.sub.i+1.epsilon.f-reachable(z)} and:

iii) if z.sub.i.epsilon.last(E1*) for some iteration subexpression E1* of E and E1 is the largest one, then let C1={z.epsilon.A(E1).vertline.z.sub.i+1.epsilon.f-reachable(z)}, C=C.orgate.C1;

iv) If z.sub.i+1.epsilon.first(E2*) for some iteration subexpression E2* of E and E2 is the largest one, then let C2={z.epsilon.A(E2).vertline.z.epsilon.f-reachable(z.sub.i)} and C=C.orgate.C2; if (z.sub.i, z.sub.i+1) is a backward edge, then let C=A(E3), wherein E3* is the largest iteration subexpression of E satisfying z.sub.i.epsilon.last(E3) and z.sub.i+1.epsilon.first(E3).

See rejection of limitations addressed in claim 4 above.

Per claim 11:

-automatically generating a required element between element pair z.sub.i and z and element pair z and z.sub.i+1 after said third element z is inserted between element pair z.sub.i and z.sub.i+1, such that said working document is effective; wherein said requirement comprises articulation points between elements z.sub.i and z (and z and z.sub.i+1) in Glushkov Automaton G; i.e., states through which all paths between z.sub.i and z (and z and z.sub.i+1) shall pass.

See rejection of limitations addressed in claim 6 above.

Per claim 12:

-automatically generating an element slot allowing user to add elements into said document, if no required element between element pair z.sub.i and z and element pair z and z.sub.i+1 is found after said third element z is inserted between element pair z.sub.i and z.sub.i+1 and if (z.sub.i, z)H ((z, z.sub.i+1)H), wherein H denotes set of edges in G; and wherein said requirement comprises articulation points between elements z.sub.i and z (and z and z.sub.i+1) in Glushkov Automaton G; i.e., states through which all paths between z.sub.1 and z (and z and z.sub.i+1) shall pass.

See rejection of limitations addressed in claim 6 above.

Per claim 13:

An XML document editor, comprising a user interface enabling user to add or delete an element into a working document, whereby said working document is converted into an XML document

file; characterized in that said XML document editor automatically generates in relating to two consecutive elements z.sub.i and z.sub.i+1 of said working document, wherein relation between said elements z.sub.i and z.sub.i+1 complies with document type definition (DTD) of said document, a list of candidate third element to be alerted to usef; wherein said third element z in said list makes relations between elements z.sub.i and z and between elements z and z.sub.i+1 complying with said DTD, after said third element z is inserted between elements z.sub.i and z.sub.i+1.

See rejection of claim 1 addressed above.

Per claim 14:

XML document editor determines whether relation between two consecutive elements comply with said DTD according to the following rule: suppose G is Glushkov Automaton of said document, z.sub.i is a state in G, 1.ltoreq.i.ltoreq.p-1, p.epsilon.N, .SIGMA.{z.sub.1, z.sub.2, . . . z.sub.p} is a sequence of states in G where z.sub.1=s, s is start state of G, z.sub.p=f, f is final state of G; if z.sub.i+1.epsilon.reachable(z.- sub.i), wherein reachable(z.sub.i) denote the set of states in G reachable from state z.sub.i, then the relation between z.sub.i and z.sub.i+1 is determined compliant with DTD of said document.

See rejection of claim 2 addressed above.

Per claim 15:

-XML generates a cell C to include said candidate third element z according to the following rule and displays said candidates in a list: suppose (z.sub.i, z.sub.i+1)H, H denotes the set of edges in

G, G is Glushkov Automaton of regular expression E corresponding to an element of said working document; further suppose .SIGMA. is a set to include states corresponding to all elements of G, A(E1) is the set of states in subexpression E1 to E, f-reachable(z.sub.i) denotes the set of states in G reachable from z.sub.i through forward edges; if z.sub.i+1.epsilon.f-reachable(z.sub.i), then let C={z.epsilon..SIGMA..ver- tline.z.epsilon.f-reachable(z.sub.i) and z.sub.i+1.epsilon.f-reachable(z)}- ; if z.sub.i+1f-reachable(z.sub.i), then let E1* be the smallest iteration subexpression of E that covers both z.sub.i and z.sub.i+1, C={z.epsilon.A(E1).vertline.z.epsilon.f-reachable(z.sub.i) or z.sub.i+1.epsilon.f-reachable(z)}. See rejection of claim 3 addressed above.

Per claim 16:

-XML generates a cell C to include said candidate third element z according to the following rule and displays said candidates in a list: suppose (z.sub.i, z.sub.i+1).epsilon.H, H denotes the set of edges in G, G is Glushkov Automaton of regular expression E corresponding to an element of said working document; further suppose .SIGMA. is a set to include states corresponding to all elements of G, A(E1) is the set of states in subexpression E1 to E, f-reachable(z.sub.i) denotes the set of states in G reachable from z.sub.i through forward edges; if (z.sub.i, z.sub.i+1) is a forward edge, let C={z.epsilon..SIGMA..vertline.z.epsilon.f-reachabl- e(z.sub.i) and z.sub.i+1.epsilon.f-reachable(z)} and: v) if z.sub.i.epsilon.last(E1*) for some iteration subexpression E1* of E and E1 is the largest one, then let C1={z.epsilon.A(E1).vertline.z.sub.i+1.ep- silon.f-reachable(z)}, C=CC1; vi) If z.sub.i+1.epsilon.first(E2*) for some iteration subexpression E2* of E and E2 is the largest one,

then let C2={zA(E2).vertline.z.epsilon.f-reachable(z.sub.i)} and C=CC2; if (z.sub.i, z.sub.i+1) is a backward edge, then let C=A(E3), wherein E3* is the largest iteration subexpression of E satisfying z.sub.i.epsilon.last(E3) and z.sub.i+1.epsilon.first(E3).

See rejection of limitations addressed in claim 4 above.

Per claim 17:

-XML document editor automatically generates a required element between element pair z.sub.i and z and element pair z and z.sub.i+1 after said third element z is inserted between element pair z.sub.i and z.sub.i+1, such that said working document is effective; wherein said requirement comprises articulation points between elements z.sub.i and z (and z and z.sub.i+1) in Glushkov Automaton G; i.e., states through which all paths between z.sub.1 and z (and z and z.sub.i+1) shall pass.

See rejection of limitations addressed in claim 5 above.

18. The XML document editor according to claim 13, wherein said XML document editor automatically generates an element slot allowing user to add elements into said document, if no required element between element pair z.sub.i and z and element pair z and z.sub.i+1 is found after said third element z is inserted between element pair z.sub.i and z.sub.i+1 and if (z.sub.i, z)H ((z, z.sub.i+1)H), wherein H denotes set of edges in G; and wherein said requirement comprises articulation points between elements z.sub.i and z (and z and z.sub.i+1) in Glushkov Automaton G; i.e., states through which all paths between z.sub.i and z (and z and z.sub.i+1) shall pass.

See rejection of limitations addressed in claim 6 above.

Conclusion

4. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Mary Steelman, whose telephone number is (571) 272-3704. The examiner can normally be reached Monday through Thursday, from 7:00 AM to 5:30 PM If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Wei Zhen can be reached at (571) 272-3708. The fax phone number for the organization where this application or proceeding is assigned: 571-273-8300.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Mary Steelman 

11/22/2006